

WHAT IS CLAIMED IS:

- 1           1.       A method for encrypting data in a computer in communication with a volatile  
2 memory and non-volatile storage device, comprising:  
3           encrypting pages in the volatile memory to move to a swap file in the non-volatile  
4 storage device as part of a virtual addressing system;  
5           moving the encrypted pages from the volatile memory to the swap file;  
6           decrypting pages in the swap file to move back into the volatile memory; and  
7           moving the decrypted pages in the swap file back into the volatile memory.
- 1           2.       The method of claim 1, further comprising:  
2           generating codes to use to encrypt and decrypt the pages.
- 1           3.       The method of claim 2, wherein the codes comprise a public/private key pair  
2 generated using a public key cryptography algorithm, wherein one key of the pair is used to  
3 encrypt the pages moved to the swap file and the other key of the pair is used to decrypt the  
4 page when moving the page from the swap file to the volatile memory.
- 1           4.       The method of claim 2, wherein the codes are permanently lost if the computer  
2 performs a boot operation.
- 1           5.       The method of claim 2, wherein the codes are loaded into a non-swappable  
2 region of the volatile memory that is not moved to the swap file.
- 1           6.       A method for encrypting files in a computer file system in communication with a  
2 volatile memory and a non-volatile storage device, wherein files in the file system are associated  
3 with groups, comprising:

4 providing, for each group, a group identifier, a list of user identifiers of users allowed to  
5 access files in the group, and a first encryption code;  
6 receiving a second encryption code for one user identifier;  
7 receiving an input/output (I/O) request from a requesting user identifier with respect to a  
8 target file, wherein one second encryption code has been received for the user identifier;  
9 determining the group associated with the target file and the first encryption code for the  
10 group;  
11 if the I/O request is a write operation, then using the determined first encryption code to  
12 encrypt the target file to write the target file to the non-volatile storage device; and  
13 if the I/O request is a read operation to read the target file from the non-volatile storage  
14 device, then performing:  
15 (i) determining whether the requesting user identifier is in the list for the  
16 determined group; and  
17 (ii) if the requesting user identifier is in the list, then using the second encryption  
18 code for the user identifier to decrypt the target file.

1 7. The method of claim 6, further comprising:  
2 for each group, generating a public and private encryption key pair using a public key  
3 encryption algorithm, wherein the first encryption code for the group is one of the generated  
4 public key or private key and the second encryption code is the other one of the public or  
5 private key generated for the group.

1 8. The method of claim 7, further comprising receiving a plurality of keys from the  
2 user, wherein each received key is used to decrypt files associated with one group identifier.

1 9. The method of claim 7, further comprising:  
2 generating an index entry in a non-swappable region in the volatile memory; and

3 adding to the index entry the user identifier of the user that provided the key, the group  
4 identifier associated with the received key, and the received key.

1 10. The method of claim 9, wherein the index entry for the user identifier and group  
2 identifier is only generated if the user identifier is included in the list associated with the group  
3 identifier, and wherein the user identifier cannot perform a read access for the target file if there  
4 is no index entry for the group identifier associated with the target file and the user identifier.

1 11. The method of claim 9, wherein files read and decrypted from the non-volatile  
2 storage device are cached in the volatile memory, and wherein if the requested file is  
3 unencrypted in the cache, returning the unencrypted file from the cache to the requesting user  
4 identifier if the requesting user identifier is in the list associated with the group identifier and  
5 there is one index entry for the user identifier and group identifier in the volatile memory.

1 12. The method of claim 1, wherein the second encryption code is accessed from a  
2 removable storage medium.

1 13. A method for encrypting files in a computer in communication with a volatile  
2 memory and non-volatile storage device, comprising:  
3 generating an encryption code to encrypt a file and a decryption code to decrypt one  
4 file encrypted with the encryption code;  
5 loading the decryption code into the volatile memory, wherein the decryption code is  
6 erased from the volatile memory when the computer reboots;  
7 encrypting files with the encryption code to transfer from the volatile memory to the  
8 non-volatile storage device; and  
9 decrypting files with the decryption code maintained in the volatile memory to transfer  
10 from the non-volatile storage device to the volatile memory.

1           14.    The method of claim 13, further comprising:  
2           generating a new encryption and decryption codes when the computer reboots, wherein  
3   the new encryption code is used to transfer files from the volatile memory to the non-volatile  
4   storage device and wherein the new decryption code is used to transfer files from the non-  
5   volatile storage device to the volatile memory as part of a read operation.

1           15.    The method of claim 13, wherein the decryption code is loaded into a non-  
2   swappable region of the volatile memory.

1           16.    The method of claim 13, wherein the files are transferred between the volatile  
2   memory and non-volatile storage as part of a virtual memory paging operation.

1           17.    A system for encrypting data, comprising:  
2           a volatile memory;  
3           a non-volatile storage device, wherein data is capable of being transferred between the  
4   volatile memory and non-volatile storage device;  
5           means for encrypting pages in the volatile memory to move to a swap file in the non-  
6   volatile storage device as part of a virtual addressing system;  
7           means for moving the encrypted pages from the volatile memory to the swap file;  
8           means for decrypting pages in the swap file to move back into the volatile memory; and  
9           means for moving the decrypted pages in the swap file back into the volatile memory.

1           18.    The system of claim 17, further comprising:  
2           means for generating codes to use to encrypt and decrypt the pages.

1           19.    The system of claim 18, wherein the codes comprise a public/private key pair  
2   generated using a public key cryptography algorithm, wherein one key of the pair is used to

3 encrypt the pages moved to the swap file and the other key of the pair is used to decrypt the  
4 page when moving the page from the swap file to the volatile memory.

1 20. The system of claim 18, wherein the codes are permanently lost if the computer  
2 performs a boot operation.

1 21. The system of claim 18, further comprising:  
2 means for loading the codes into a non-swappable region of the volatile memory that is  
3 not moved to the swap file.

1 22. A system for encrypting files, comprising:  
2 a non-volatile storage device, wherein the non-volatile storage device includes a  
3 computer file system, wherein files in the file system are associated with groups;  
4 means for providing, for each group, a group identifier, a list of user identifiers of users  
5 allowed to access files in the group, and a first encryption code;  
6 means for receiving a second encryption code for one user identifier;  
7 means for receiving an input/output (I/O) request from a requesting user identifier with  
8 respect to a target file, wherein one second encryption code has been received for the user  
9 identifier;  
10 means for determining the group associated with the target file and the first encryption  
11 code for the group;  
12 means for using the determined first encryption code to encrypt the target file to write  
13 the target file to the non-volatile storage device if the I/O request is a write operation; and  
14 means for performing if the I/O request is a read operation to read the target file from  
15 the non-volatile storage device:  
16 (i) determining whether the requesting user identifier is in the list for the  
17 determined group; and

18 (ii) if the requesting user identifier is in the list, then using the second encryption  
19 code for the user identifier to decrypt the target file.

1 23. The system of claim 22, further comprising:  
2 means for generating, for each group, a public and private encryption key pair using a  
3 public key encryption algorithm, wherein the first encryption code for the group is one of the  
4 generated public key or private key and the second encryption code is the other one of the  
5 public or private key generated for the group.

1 24. The system of claim 23, further comprising:  
2 means for receiving a plurality of keys from the user, wherein each received key is used  
3 to decrypt files associated with one group identifier.

1 25. The system of claim 23, further comprising:  
2 means for generating an index entry in a non-swappable region in the volatile memory;  
3 and  
4 means for adding to the index entry the user identifier of the user that provided the key,  
5 the group identifier associated with the received key, and the received key.

1 26. The system of claim 25, wherein the index entry for the user identifier and group  
2 identifier is only generated if the user identifier is included in the list associated with the group  
3 identifier, and wherein the user identifier cannot perform a read access for the target file if there  
4 is no index entry for the group identifier associated with the target file and the user identifier.

1 27. The system of claim 25, wherein files read and decrypted from the non-volatile  
2 storage device are cached in the volatile memory, further comprising:

3           returning the unencrypted file from the cache to the requesting user identifier if the  
4 requested file is unencrypted in the cache and if the requesting user identifier is in the list  
5 associated with the group identifier and if there is one index entry for the user identifier and  
6 group identifier in the volatile memory.

1           28.    The system of claim 22, wherein the second encryption code is accessed from  
2 a removable storage medium.

1           29.    A system for encrypting files, comprising:  
2 a volatile memory;  
3 a non-volatile storage device, wherein data is capable of being transferred between the  
4 volatile memory and non-volatile storage device;  
5 means for generating an encryption code to encrypt a file and a decryption code to  
6 decrypt one file encrypted with the encryption code;  
7 means for loading the decryption code into the volatile memory, wherein the decryption  
8 code is erased from the volatile memory when the computer reboots;  
9 means for encrypting files with the encryption code to transfer from the volatile memory  
10 to the non-volatile storage device; and  
11 means for decrypting files with the decryption code maintained in the volatile memory to  
12 transfer from the non-volatile storage device to the volatile memory.

1           30.    The system of claim 29, further comprising:  
2 means for generating a new encryption and decryption codes when the computer  
3 reboots, wherein the new encryption code is used to transfer files from the volatile memory to  
4 the non-volatile storage device and wherein the new decryption code is used to transfer files  
5 from the non-volatile storage device to the volatile memory as part of a read operation.

1           31.     The system of claim 29, wherein the decryption code is loaded into a non-  
2 swappable region of the volatile memory.

1           32.     The system of claim 29, wherein the files are transferred between the volatile  
2 memory and non-volatile storage as part of a virtual memory paging operation.

1           33.     An article of manufacture including program logic for encrypting data in a  
2 computer in communication with a volatile memory and non-volatile storage device, by:  
3           encrypting pages in the volatile memory to move to a swap file in the non-volatile  
4 storage device as part of a virtual addressing system;  
5           moving the encrypted pages from the volatile memory to the swap file;  
6           decrypting pages in the swap file to move back into the volatile memory; and  
7           moving the decrypted pages in the swap file back into the volatile memory.

1           34.     The article of manufacture of claim 33, further comprising:  
2           generating codes to use to encrypt and decrypt the pages.

1           35.     The article of manufacture of claim 34, wherein the codes comprise a  
2 public/private key pair generated using a public key cryptography algorithm, wherein one key  
3 of the pair is used to encrypt the pages moved to the swap file and the other key of the pair is  
4 used to decrypt the page when moving the page from the swap file to the volatile memory.

1           36.     The article of manufacture of claim 34, wherein the codes are permanently lost  
2 if the computer performs a boot operation.

1           37.     The article of manufacture of claim 34, wherein the codes are loaded into a  
2 non-swappable region of the volatile memory that is not moved to the swap file.

1           38.     An article of manufacture including program logic for encrypting files in a  
2 computer file system in communication with a volatile memory and a non-volatile storage  
3 device, wherein files in the file system are associated with groups by:  
4           providing, for each group, a group identifier, a list of user identifiers of users allowed to  
5 access files in the group, and a first encryption code;  
6           receiving a second encryption code for one user identifier;  
7           receiving an input/output (I/O) request from a requesting user identifier with respect to a  
8 target file, wherein one second encryption code has been received for the user identifier;  
9           determining the group associated with the target file and the first encryption code for the  
10 group;  
11           if the I/O request is a write operation, then using the determined first encryption code to  
12 encrypt the target file to write the target file to the non-volatile storage device; and  
13           if the I/O request is a read operation to read the target file from the non-volatile storage  
14 device, then performing:  
15                 (i) determining whether the requesting user identifier is in the list for the  
16 determined group; and  
17                 (ii) if the requesting user identifier is in the list, then using the second encryption  
18 code for the user identifier to decrypt the target file.

1           39.     The article of manufacture of claim 38, further comprising:  
2           for each group, generating a public and private encryption key pair using a public key  
3 encryption algorithm, wherein the first encryption code for the group is one of the generated  
4 public key or private key and the second encryption code is the other one of the public or  
5 private key generated for the group.

1           40.    The article of manufacture of claim 39, further comprising receiving a plurality  
2 of keys from the user, wherein each received key is used to decrypt files associated with one  
3 group identifier.

1           41.    The article of manufacture of claim 39, further comprising:  
2           generating an index entry in a non-swappable region in the volatile memory; and  
3           adding to the index entry the user identifier of the user that provided the key, the group  
4 identifier associated with the received key, and the received key.

1           42.    The article of manufacture of claim 41, wherein the index entry for the user  
2 identifier and group identifier is only generated if the user identifier is included in the list  
3 associated with the group identifier, and wherein the user identifier cannot perform a read  
4 access for the target file if there is no index entry for the group identifier associated with the  
5 target file and the user identifier.

1           43.    The article of manufacture of claim 41, wherein files read and decrypted from  
2 the non-volatile storage device are cached in the volatile memory, and wherein if the requested  
3 file is unencrypted in the cache, returning the unencrypted file from the cache to the requesting  
4 user identifier if the requesting user identifier is in the list associated with the group identifier and  
5 there is one index entry for the user identifier and group identifier in the volatile memory.

1           44.    The article of manufacture of claim 38, wherein the second encryption code is  
2 accessed from a removable storage medium.

1           45.    An article of manufacture including program logic for encrypting files in a  
2 computer in communication with a volatile memory and non-volatile storage device by:

3 generating an encryption code to encrypt a file and a decryption code to decrypt one file  
4 encrypted with the encryption code;  
5 loading the decryption code into the volatile memory, wherein the decryption code is  
6 erased from the volatile memory when the computer reboots;  
7 encrypting files with the encryption code to transfer from the volatile memory to the  
8 non-volatile storage device; and  
9 decrypting files with the decryption code maintained in the volatile memory to transfer  
10 from the non-volatile storage device to the volatile memory.

1 46. The article of manufacture of claim 45, further comprising:  
2 generating a new encryption and decryption codes when the computer reboots, wherein  
3 the new encryption code is used to transfer files from the volatile memory to the non-volatile  
4 storage device and wherein the new decryption code is used to transfer files from the non-  
5 volatile storage device to the volatile memory as part of a read operation.

1 47. The article of manufacture of claim 45, wherein the decryption code is loaded  
2 into a non-swappable region of the volatile memory.

1 48. The article of manufacture of claim 45, wherein the files are transferred between  
2 the volatile memory and non-volatile storage as part of a virtual memory paging operation.